

# JAVA.UTIL.LOCALE CLASS

[http://www.tutorialspoint.com/java/util/java\\_util\\_locale.htm](http://www.tutorialspoint.com/java/util/java_util_locale.htm)

Copyright © tutorialspoint.com

## Introduction

The **java.util.Locale** class object represents a specific geographical, political, or cultural region. .Following are the important points about Locale:

- An operation that requires a Locale to perform its task is called locale-sensitive and uses the Locale to form information for the user.
- Locale is a mechanism for identifying objects, not a container for the objects themselves.

## Class declaration

Following is the declaration for **java.util.Locale** class:

```
public final class Locale
    extends Object
        implements Cloneable, Serializable
```

## Field

Following are the fields for **java.util.Locale** class:

- **static Locale CANADA** -- This is the constant for country.
- **static Locale CANADA FRENCH** -- This is the constant for country.
- **static Locale CHINA** -- This is the constant for country.
- **static Locale CHINESE** -- This is the constant for language.
- **static Locale ENGLISH** -- This is the constant for language.
- **static Locale FRANCE** -- This is the constant for country.
- **static Locale FRENCH** -- This is the constant for language.
- **static Locale GERMAN** -- This is the constant for language.
- **static Locale GERMANY** -- This is the constant for country.
- **static Locale ITALIAN** -- This is the constant for language.
- **static Locale ITALY** -- This is the constant for country.
- **static Locale JAPAN** -- This is the constant for country.
- **static Locale JAPANESE** -- This is the constant for language.
- **static Locale KOREA** -- This is the constant for country.
- **static Locale KOREAN** -- This is the constant for language.

- **static Locale PRC** -- This is the constant for country.
- **static Locale ROOT** -- This is the constant for root locale.
- **static Locale SIMPLIFIED CHINESE** -- This is the constant for language.
- **static Locale TAIWAN** -- This is the constant for country.
- **static Locale TRADITIONAL CHINESE** -- This is the constant for language.
- **static Locale UK** -- This is the constant for country.
- **static Locale US** -- This is the constant for country.

## Class constructors

S.N.	Constructor & Description
1	<b>Locale(String language)</b> This constructs a locale from a language code.
2	<b>Locale(String language, String country)</b> This constructs a locale from a language code.
3	<b>Locale(String language, String country, String variant)</b> This constructs a locale from language, country, variant.

## Class methods

S.N.	Method & Description
1	<a href="#"><u>Object clone()</u></a> This method overrides Cloneable
2	<a href="#"><u>boolean equals(Object obj)</u></a> This method returns true if this Locale is equal to another object.
3	<a href="#"><u>static Locale[] getAvailableLocales()</u></a> This method returns an array of all installed locales.
4	<a href="#"><u>String getCountry()</u></a> This method returns the country/region code for this locale, which will either be the empty string or an uppercase ISO 3166 2-letter code.
5	<a href="#"><u>static Locale getDefault()</u></a> This method gets the current value of the default locale for this instance of the Java Virtual Machine.
6	<a href="#"><u>String getDisplayCountry()</u></a> This method returns a name for the locale's country that is appropriate for display to the user.
7	<a href="#"><u>S String getDisplayCountry(Locale inLocale)</u></a> This method Returns a name for the locale's country that is appropriate for display to the user.

8	<a href="#"><u>String getDisplayLanguage()</u></a> This method returns a name for the locale's language that is appropriate for display to the user.
9	<a href="#"><u>String getDisplayLanguage(Locale inLocale)</u></a> This method returns a name for the locale's language that is appropriate for display to the user.
10	<a href="#"><u>String getDisplayName()</u></a> This method returns a name for the locale that is appropriate for display to the user.
11	<a href="#"><u>String getDisplayName(Locale inLocale)</u></a> This method returns a name for the locale that is appropriate for display to the user.
12	<a href="#"><u>String getDisplayVariant()</u></a> This method returns a name for the locale's variant code that is appropriate for display to the user.
13	<a href="#"><u>String getDisplayVariant(Locale inLocale)</u></a> This method returns a name for the locale's variant code that is appropriate for display to the user.
14	<a href="#"><u>String getISO3Country()</u></a> This method returns a three-letter abbreviation for this locale's country.
15	<a href="#"><u>String getISO3Language()</u></a> This method returns returns a three-letter abbreviation for this locale's language.
16	<a href="#"><u>static String[] getISOCountries()</u></a> This method returns a list of all 2-letter country codes defined in ISO 3166.
17	<a href="#"><u>static String[] getISOLanguages()</u></a> This method returns a list of all 2-letter language codes defined in ISO 639.
18	<a href="#"><u>String getLanguage()</u></a> This method returns the language code for this locale, which will either be the empty string or a lowercase ISO 639 code.
19	<a href="#"><u>String getVariant()</u></a> This method returns the variant code for this locale.
20	<a href="#"><u>int hashCode()</u></a> This method override hashCode.
21	<a href="#"><u>static void setDefault(Locale newLocale)</u></a> This method sets the default locale for this instance of the Java Virtual Machine.
22	<a href="#"><u>String toString()</u></a> This method is the getter for the programmatic name of the entire locale, with the language, country and variant separated by underbars.

## Methods inherited

This class inherits methods from the following classes:

- java.util.Object